Adaptive H^{∞} Control of Nonlinear Systems with Neural Networks

JIANG Chang-sheng, CHEN Mou
(*Automation College, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China*)

Abstract: The discussion is devoted to the adaptive H^{∞} control method based on RBF neural networks for uncertain nonlinear systems in this paper. The controller consists of an equivalent controller and an H^{∞} controller. The RBF neural networks are used to approximate the nonlinear functions and the approximation errors of the neural networks are used in the adaptive law to improve the performance of the systems. The H^{∞} controller is designed for attenuating the influence of external disturbance and neural network approximation errors. The controller can not only guarantee stability of the nonlinear systems, but also attenuate the effect of the external disturbance and neural networks approximation errors to reach performance indexes. Finally, an example validates the effectiveness of this method.

Key words: neural networks; nonlinear systems; adaptive control; H^{∞} control

基于 **RBF** 神经网络的一类不确定非线性系统自适应 H^{∞}控制. 姜长生, 陈谋, 中国航空学报, 2003, 16(1): 36-41.

摘 要: 基于 RBF 神经网络提出了一种 H^{∞}自适应控制方法。控制器由等效控制器和 H^{∞}控制器两部分组成。用 RBF 神经网络逼近非线性函数, 并把逼近误差引入到网络权值的自适应律中用以改善系统的动态性能。H^{∞}控制器用于减弱外部及神经网络的逼近误差对跟踪的影响。所设计的控制器不仅保证了闭环系统的稳定性, 而且使外部干扰及神经网络的逼近误差对跟踪的影响减小到给定的性能指标。最后给出的算例验证了该方法的有效性。

关键词: 神经网络; 非线性系统; 自适应控制; H^{∞}控制
文章编号: 1000-9361(2003)01-0036-06 中图分类号: V245.3 文献标识码: A

Control of uncertain nonlinear systems is an active subject in the modern control area and there are a lot of research results about them. Neural networks(NN) are widely used to control nonlinear systems, so a lot of control methods were proposed[1-4]. But most of them are control methods of single input and single output (SISO) nonlinear systems. There are few control methods of multi-input and multi-output (MIMO) uncertain nonlinear systems. It is very significant that control methods of MIMO nonlinear systems are studied.

Recently, combining adaptive control and robust control to control nonlinear systems has received increasing attention for using both advantages of the adaptive control and robust control. Ref. [5] has studied a stable control method about MIMO nonlinear systems based on BP neural networks. But it does not consider external uncertainty and modeling error. Simultaneously, the controller lacks the ability to diminish the disturbance and to counteract the modeling error of the NN. Ref. [6] has studied a control method based on fuzzy control for multivariable nonlinear systems, but the dynamic performance of a nonlinear system is not good.

This paper combines RBF neural networks with adaptive H^{∞} Control to propose a new control method about MIMO nonlinear systems. This method divides the uncertainty of a nonlinear system into two parts. One is the error of the NN and the other is the external uncertainty and modeling error. The RBF neural networks are used to ap-

Received date: 2002-06-25; Revision received date: 2002-11-20
Foundation item: Nation Natural Science Foundation of China(60174045); Aeronautical Science Foundation of China(01D52025)
Article URL: http://www.hkxb.net.cn/cja/2003/01/0036/

proximate the nonlinear functions; the approximate errors of the neural networks are used in the adaptive law to improve performance of the systems. The H^{∞} controller is designed for attenuating the external disturbance and neural networks approximation errors. The designed controller can not only guarantee stability of the whole nonlinear system, but also attenuate the influence of the external disturbance and neural network approximation errors to reach a prescript level.

1 Problem Formulation

Considering the affine uncertainty nonlinear system

$$\left.\begin{aligned} \dot{\boldsymbol{x}} &= \alpha(\boldsymbol{x}) + \beta(\boldsymbol{x})\boldsymbol{u} + \gamma(\boldsymbol{x}) \\ \boldsymbol{y} &= \boldsymbol{h}(\boldsymbol{x}) \end{aligned}\right\} \quad (1)$$

where $\boldsymbol{x} = (x_1 \ \cdots \ x_n)^{\mathrm{T}} \in \mathbf{R}^n$; $\alpha(\boldsymbol{x}) = [\alpha_1(\boldsymbol{x}) \ \cdots \ \alpha_n(\boldsymbol{x})]^{\mathrm{T}} \in \mathbf{R}^n$ are smooth nonlinear vector functions; $\beta(\boldsymbol{x}) = [\beta_{ij}(\boldsymbol{x})] \in \mathbf{R}^{n\times m}$ is the control gain matrix; $\boldsymbol{u} \in \mathbf{R}^m$ and $\gamma(\boldsymbol{x}) = [\gamma_1(\boldsymbol{x}) \ \cdots \ \gamma_n(\boldsymbol{x})]^{\mathrm{T}} \in \mathbf{R}^n$ are smooth nonlinear function vectors denoting the disturbance and the modeling uncertainties of the system.

Suppose that the nonlinear system satisfies the following assumptions[1-4]:

Assumption: A1: The relative degree of the system is $\{r_1, \cdots, r_m\}$, and $r_1 + \cdots + r_m = n$;

A2: $L_{\alpha}L_{\gamma}^{k}h_i(\boldsymbol{x}) = 0$, $L_{\gamma}L_{\alpha}^{k}h_i(\boldsymbol{x}) = 0$, and $k = 1, \cdots, r_i - 2$, $i = 1, \cdots, m$.

A3: The model error $\gamma(\boldsymbol{x})$ has a certain expression.

Choosing the coordinate transform $\boldsymbol{Z} = \Phi(\boldsymbol{x})$ based on above assumptions, then the nonlinear system can be transformed into a new standard form[6]

$$\left.\begin{aligned} &\dot{\varphi}_1 = \varphi_2, \cdots, \dot{\varphi}_{r_1-1} = \varphi_{r_1} \\ &\dot{\varphi}_{r_1} = f_1(\boldsymbol{x}) + g_{11}(\boldsymbol{x})u_1 + \cdots + \\ &\quad g_{1m}(\boldsymbol{x})u_m + d_1(\boldsymbol{x}), \\ &y_1 = \varphi_1, \cdots \\ &\dot{\phi}_1 = \phi_2, \cdots, \dot{\phi}_{r_m-1} = \phi_{r_m} \\ &\dot{\phi}_{r_m} = f_m(\boldsymbol{x}) + g_{m1}(\boldsymbol{x})u_1 + \cdots + \\ &\quad g_{mm}(\boldsymbol{x})u_m + d_m(\boldsymbol{x}) \\ &y_m = \phi_1 \end{aligned}\right\} \quad (2)$$

where $f_i(\boldsymbol{x}) = L_{\alpha}^{r_i}h_i(\boldsymbol{x})$, $i = 1, \cdots, m$; $\boldsymbol{x} = \Phi^{-1}(\boldsymbol{Z})$; $g_{ij}(\boldsymbol{x}) = L_{\beta_{ij}}L_{\alpha}^{r_i-1}h_i(\boldsymbol{x})$, $i, j = 1, \cdots, m$; $d_i(\boldsymbol{x}) = L_{\gamma}^{r_i}h_i(\boldsymbol{x})$, $i = 1, \cdots, m$; $\boldsymbol{y} = [\varphi_1 \ \psi_1 \ \cdots \ \phi_1]^{\mathrm{T}}$. So Eq. (2) becomes

$$\begin{bmatrix} y_1^{(r_1)} \\ \vdots \\ y_m^{(r_m)} \end{bmatrix} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{G}(\boldsymbol{x})\begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} + \boldsymbol{d}(\boldsymbol{x}) \quad (3)$$

where

$$\boldsymbol{G}(\boldsymbol{x}) = \begin{bmatrix} g_{11}(\boldsymbol{x}) & \cdots & g_{1m}(\boldsymbol{x}) \\ \vdots & \cdots & \vdots \\ g_{m1}(\boldsymbol{x}) & \cdots & g_{mm}(\boldsymbol{x}) \end{bmatrix}$$

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} f_1(\boldsymbol{x}) \\ \vdots \\ f_m(\boldsymbol{x}) \end{bmatrix}, \quad \boldsymbol{d}(\boldsymbol{x}) = \begin{bmatrix} d_1(\boldsymbol{x}) \\ \vdots \\ d_m(\boldsymbol{x}) \end{bmatrix}$$

Suppose that Eq. (3) satisfies the following assumptions[1-5]:

B1: In a compact set, $\boldsymbol{G}(\boldsymbol{x})$ is nonsingular and has a bounded norm, and satisfies the following condition

$$\sigma_G(\boldsymbol{G}(\boldsymbol{x})) \geq b > 0, \ \forall \boldsymbol{x} \in S \quad (4)$$

where σ_G represents the smallest singular value of the matrix $\boldsymbol{G}(\boldsymbol{x})$; b is an arbitrary constant and S is a compact set.

B2: $\boldsymbol{f}(\boldsymbol{x})$ and $\boldsymbol{d}(\boldsymbol{x})$ are smooth vector functions, and $\boldsymbol{d}(\boldsymbol{x})$ has bounds.

The control goal is to design an adaptation robust neural network controller such that the output y_i of the nonlinear system and the $r_i - 1$ derivative of the output track the reference signal.

2 Design of Adaptive H^{∞} Controller for Nonlinear Systems Based on RBF Neural Networks

Constructing the controller

$$\boldsymbol{u} = \boldsymbol{u}_c + \boldsymbol{u}_h \quad (5)$$

where

$$\left.\begin{aligned} \boldsymbol{u}_c &= \hat{\boldsymbol{G}}^{-1}(\boldsymbol{x}\backslash \boldsymbol{w}_g)[-\hat{f}(\boldsymbol{x}\backslash \boldsymbol{w}_f) + \tau], \\ \boldsymbol{u}_h &= -\hat{\boldsymbol{G}}^{-1}(\boldsymbol{x}\backslash \boldsymbol{w}_g)\boldsymbol{u}_a \end{aligned}\right\} \quad (6)$$

$\boldsymbol{u}_c$ is the equivalent controller; $\boldsymbol{u}_h$ is a compensatory controller which is used to overcome the influence of the compound disturbance and to increase robustness.

where $\boldsymbol{u} = [u_1 \ \cdots \ u_m]^{\mathrm{T}}$, $\tau = [\tau_1 \ \cdots \ \tau_m]^{\mathrm{T}}$

$$\hat{\boldsymbol{G}}(\boldsymbol{x}\backslash\boldsymbol{w}_g)=\begin{bmatrix}\hat{g}_{11}(\boldsymbol{x}\backslash\boldsymbol{w}_{g_{11}}) & \cdots & \hat{g}_{1m}(\boldsymbol{x}\backslash\boldsymbol{w}_{g_{1m}})\\ \vdots & \cdots & \vdots\\ \hat{g}_{m1}(\boldsymbol{x}\backslash\boldsymbol{w}_{g_{m1}}) & \cdots & \hat{g}_{mm}(\boldsymbol{x}\backslash\boldsymbol{w}_{g_{mm}})\end{bmatrix}$$

$$\hat{\boldsymbol{f}}(\boldsymbol{x}\backslash\boldsymbol{w}_f)=[\hat{f}_1(\boldsymbol{x}\backslash\boldsymbol{w}_{f1}),\cdots,\hat{f}_m(\boldsymbol{x}\backslash\boldsymbol{w}_{fm})]^{\mathrm{T}}$$

$\hat{\boldsymbol{f}}(\boldsymbol{x}\backslash\boldsymbol{w}_f)$ and $\hat{\boldsymbol{G}}(\boldsymbol{x}\backslash\boldsymbol{w}_g)$ are the approximation to $\boldsymbol{f}(\boldsymbol{x})$ and $\boldsymbol{G}(\boldsymbol{x})$, used by the neural networks. Determining

$$\left.\begin{aligned}\tau_1&=y_{1d}^{(r_1)}+\lambda_{1r_1}(y_{1d}^{(r_1-1)}-y_1^{(r_1-1)})+\cdots\\&\quad+\lambda_{11}(y_{1d}-y_1)\\&\ \ \vdots\\ \tau_m&=y_{md}^{(r_m)}+\lambda_{mr_m}(y_{md}^{(r_m-1)}-y_1^{(r_m-1)})+\cdots\\&\quad+\lambda_{m1}(y_{md}-y_m)\end{aligned}\right\}\tag{7}$$

where $\lambda_{ij}(j=1,\cdots,r_i)$ satisfies the condition that roots of $h_i(s)=s^n+\lambda_{1r_i}s^{n-1}+\cdots+\lambda_{i1}=0$ lie in the left plane.

So Eq. (3) becomes

$$\begin{bmatrix}y_1^{(r_1)}\\ \vdots\\ y_m^{(r_m)}\end{bmatrix}=\begin{bmatrix}f_1(\boldsymbol{x})-\hat{f}_1(\boldsymbol{x}\backslash\boldsymbol{w}_{f1})\\ \vdots\\ f_m(\boldsymbol{x})-\hat{f}_m(\boldsymbol{x}\backslash\boldsymbol{w}_{fm})\end{bmatrix}+(\boldsymbol{G}(\boldsymbol{x})-\hat{\boldsymbol{G}}(\boldsymbol{x}\backslash\boldsymbol{w}_g))\begin{bmatrix}u_1\\ \vdots\\ u_m\end{bmatrix}+\begin{bmatrix}\tau_1\\ \vdots\\ \tau_m\end{bmatrix}-\begin{bmatrix}u_{1a}\\ \vdots\\ u_{ma}\end{bmatrix}+\begin{bmatrix}d_1(\boldsymbol{x})\\ \vdots\\ d_2(\boldsymbol{x})\end{bmatrix}\tag{8}$$

Determining $e_i=y_i-y_{id}$, $i=1,\cdots,m$, then Eq. (8) becomes

$$\begin{bmatrix}e_1^{(r_1)}+\lambda_{1r_1}e_1^{(r_1-1)}+\cdots+\lambda_{11}e_1\\ \vdots\\ e_m^{(r_m)}+\lambda_{1r_m}e_m^{(r_m-1)}+\cdots+\lambda_{m1}e_m\end{bmatrix}=\begin{bmatrix}\hat{f}_1(\boldsymbol{x}\backslash\boldsymbol{w}_{f1})-f_1(\boldsymbol{x})\\ \vdots\\ \hat{f}_m(\boldsymbol{x}\backslash\boldsymbol{w}_{fm})-f_m(\boldsymbol{x})\end{bmatrix}+(\hat{\boldsymbol{G}}(\boldsymbol{x}\backslash\boldsymbol{w}_g)-\boldsymbol{G}(\boldsymbol{x}))\begin{bmatrix}u_1\\ \vdots\\ u_m\end{bmatrix}+\begin{bmatrix}u_{1a}\\ \vdots\\ u_{ma}\end{bmatrix}-\begin{bmatrix}d_1(\boldsymbol{x})\\ \vdots\\ d_m(\boldsymbol{x})\end{bmatrix}\tag{9}$$

First, one must design the adaptive robust control law of each subsystem based on the neural network and then consider the stability of the whole system.

The i-th subsystem is as

$$e_i^{(r_i)}+\lambda_{ir_i}e_i^{(r_i-1)}+\cdots+\lambda_{1}e_i=\hat{f}_i(\boldsymbol{x}\backslash\boldsymbol{w}_{fi})-f_i(\boldsymbol{x})+\Delta\boldsymbol{G}_i\boldsymbol{u}+u_{ia}-d_i(\boldsymbol{x})\tag{10}$$

where $\Delta\boldsymbol{G}_i=[\hat{g}_{i1}(\boldsymbol{x}\backslash\boldsymbol{w}_{g_{i1}})-g_{i1}(\boldsymbol{x}),\cdots,\hat{g}_{im}(\boldsymbol{x}\backslash\boldsymbol{w}_{g_{im}})-g_{im}(\boldsymbol{x})]$

Determining $\boldsymbol{e}_{ip}=[e_i\ \cdots\ e_i^{(r_i-1)}]^{\mathrm{T}}$, so the above equation is written as

$$\mathring{\boldsymbol{e}}_{ip}=\boldsymbol{A}_i\boldsymbol{e}_{ip}+\boldsymbol{B}_iu_{ia}+\boldsymbol{B}_i[f_i(\boldsymbol{x})-\hat{f}_i(\boldsymbol{x}\backslash\boldsymbol{w}_{fi})+\Delta\boldsymbol{G}_i\boldsymbol{u}-d_i(\boldsymbol{x})]\tag{11}$$

where

$$\boldsymbol{A}_i=\begin{bmatrix}0&1&0&\cdots&0\\0&0&1&\cdots&0\\ \vdots&\vdots&\vdots&\vdots&\vdots\\ -\lambda_{i1}&-\lambda_{i2}&\cdots&-\lambda_{i(r_i-1)}&-\lambda_{ir_i}\end{bmatrix}$$

$$\boldsymbol{B}_i=\begin{bmatrix}0\\0\\ \vdots\\1\end{bmatrix}$$

Now, the design task is to find the controller u_{ia} and the adaptive law of $\boldsymbol{w}_{fi}$ and $\boldsymbol{w}_{g_{ij}}$, $j=1,\cdots,m$.

Determining

$$\begin{cases}\boldsymbol{w}_{fi}^{*}=\arg\min\limits_{\boldsymbol{w}_{fi}\in\Omega_{fi}}\left[\sup\limits_{\boldsymbol{x}\in\boldsymbol{u}_c}\left|\hat{f}_i(\boldsymbol{x}\backslash\boldsymbol{w}_{fi})-f_i(\boldsymbol{x})\right|\right]\\ \boldsymbol{w}_{g_{ij}}^{*}=\arg\min\limits_{\boldsymbol{w}_{g_{ij}}\in\Omega_{g_{ij}}}\left[\sup\limits_{\boldsymbol{x}\in\boldsymbol{u}_c}\left|\hat{g}_{ij}(\boldsymbol{x}\backslash\boldsymbol{w}_{g_{ij}})-g_{ij}(\boldsymbol{x})\right|\right]\end{cases}\tag{12}$$

where $\Omega_{fi}=\{\boldsymbol{w}_{fi}:\|\boldsymbol{w}_{fi}\|\le M_{fi}\}$, $\Omega_{g_{ij}}=\{\boldsymbol{w}_{g_{ij}}:\|\boldsymbol{w}_{g_{ij}}\|\le M_{g_{ij}}\}$ are the valid field of the parameters; M_{fi} and $M_{g_{ij}}$ are designed parameters. The definition of the least approximation error of the neural network is

$$\omega_i=(\hat{f}_i(\boldsymbol{x}\backslash\boldsymbol{w}_{fi}^{*})-f_i(\boldsymbol{x}))+\Delta\boldsymbol{G}^{*}\boldsymbol{u}\tag{13}$$

where $\Delta\boldsymbol{G}_i^{*}=[\hat{g}_{i1}(\boldsymbol{x}\backslash\boldsymbol{w}_{g_{i1}}^{*})-g_{i1}(\boldsymbol{x}),\cdots,\hat{g}_{im}(\boldsymbol{x}\backslash\boldsymbol{w}_{g_{im}}^{*})-g_{im}(\boldsymbol{x})]$

So Eq. (11) becomes

$$\mathring{\boldsymbol{e}}_{ip}=\boldsymbol{A}\boldsymbol{e}_{ip}+\boldsymbol{B}_iu_{ia}+\boldsymbol{B}_i[(\hat{f}_i(\boldsymbol{x}\backslash\boldsymbol{w}_{fi})-\hat{f}_i(\boldsymbol{x}\backslash\boldsymbol{w}_{fi}^{*}))+(\Delta\boldsymbol{G}_i-\Delta\boldsymbol{G}_i^{*})\boldsymbol{u}]+\boldsymbol{B}_i[\omega_i-d_i(\boldsymbol{x})]\tag{14}$$

Suppose that $\hat{f}_i(\cdot)$ and $\hat{g}_{ij}(\cdot)$ are RBF neural networks, and then

$$\hat{f}_i(\boldsymbol{x}\backslash\boldsymbol{w}_{fi})=\boldsymbol{w}_{fi}^{\mathrm{T}}\Phi(\boldsymbol{x}),\ \hat{g}_{ij}(\boldsymbol{x}\backslash\boldsymbol{w}_{g_{ij}})=\boldsymbol{w}_{g_{ij}}^{\mathrm{T}}\Phi_j(\boldsymbol{x})\tag{15}$$

Moreover, RBF neural networks have the following form

$$y_i = \sum_{j=1}^{m} w_j \Phi_j(\xi) = w^{T}\Phi(\xi) \qquad (16)$$

where $w = [w_1 \quad \cdots \quad w_m]^{T}$ is the weight value of the neural networks; $\Phi(\xi) = [\phi_1(\xi) \quad \cdots \quad \phi_m(\xi)]^{T}$, $\phi_i(\xi)$ is the base function. So Eq. (14) can be written as

$$\mathring{e}_{ip} = A_i e_{ip} + B_i u_{ia} + B_i[\tilde{w}_{fi}^{T}\Phi(x) + \tilde{w}_{g_i}^{T} u] + B_i\omega \qquad (17)$$

where $\omega = \omega_i - d_i$, $\tilde{w}_{fi} = w_{fi} - w_{fi}^{*}$, $\tilde{w}_{g_i} = [(w_{g_{i1}} - w_{g_{i1}}^{*}), \cdots, (w_{g_{im}} - w_{g_{im}}^{*})]$.

and ω is the compound disturbance.

H^{∞} controller can be written as

$$u_{ia} = -\frac{1}{\gamma} B_i^{T} p_i e_{ip} \qquad (18)$$

where γ is the design parameter.

The adaptation law can be chosen as

$$\mathring{w}_{fi} = \begin{cases} -\mu_i e_{ip}^{T} p_i B_i \Phi(x) + \epsilon(\|w_{fi}\| - M_{fi}), \\ \qquad \|w_{fi}\| \le M_f,\ e_{ip}^{T} p_i B_i w_{fi}^{T}\Phi(x) > 0 \\ P_{r_1}[\bullet], \\ \qquad \|w_{fi}\| = M_f,\ e_{ip}^{T} p_i B_i w_{fi}^{T}\Phi(x) < 0 \end{cases} \qquad (19)$$

$$\mathring{w}_{g_{ij}} = \begin{cases} -\rho_{ij} e_{ip}^{T} p_i B_i \Phi_j(x) + \sigma_{ij}(\|w_{g_{ij}}\| - M_{g_{ij}}), \\ \qquad \|w_{g_{ij}}\| \le M_{g_{ij}},\ e_{ip}^{T} p_i B_i w_{g_{ij}}^{T}\Phi_j(x) > 0 \\ P_{r_2}[\bullet], \\ \qquad \|w_{g_{ij}}\| = M_{g_{ij}},\ e_{ip}^{T} p_i B_i w_{g_{ij}}^{T}\Phi_j(x) < 0 \end{cases} \qquad (20)$$

where

$$P_{r_1}[\bullet] = -\mu_i e_{ip}^{T} p_i B_i \Phi(x) + \mu_i \frac{e_{ip}^{T} p_i B_i w_{fi} w_{fi}^{T}\Phi(x)}{\|w_{fi}\|^{2}} \qquad (21)$$

$$P_{r_2}[\bullet] = -\rho_{ij} e_{ip}^{T} p_i B_i \Phi_j(x) + \rho_{ij} \frac{e_{ip}^{T} p_i B_i w_{g_{ij}} w_{g_{ij}}^{T}\Phi(x)}{\|w_{g_{ij}}\|^{2}} \qquad (22)$$

μ_i, ρ_i are the learning rates.

Determining the parameters ϵ and σ_{ij} as

$$\epsilon = \alpha_i \operatorname{sgn}(w_{fi}),\ \sigma_{ij} = \beta_{ij}\operatorname{sgn}(w_{g_{ij}}) \qquad (23)$$

$\alpha > 0$ and $\beta > 0$ are constant vectors to be arbitrarily chosen, which are used to adjust the dynamic performance. $p_i = p_i^{T} > 0$ is the solution of the Riccati equation

$$p_i A_i + A_i^{T} p_i + Q_i \frac{2}{\gamma} p_i B_i B_i^{T} p_i + \frac{1}{k^2} p_i B_i B_i^{T} p_i = 0,\ Q_i > 0 \qquad (24)$$

3 Performance Analysis

Theorem 1 Considering the system Eq. (3), the control is defined by Eq. (5). u_c can be chosen as Eq. (6) and the parameter adaptation law can be chosen as Eq. (19) and Eq. (20). Then

(1) In the adaptive process, there are always $\|w_{fi}\| \le M_{fi}$, $\|w_{g_{ij}}\| \le M_{g_{ij}}$, $(1 \le i \le m,\ 1 \le j \le m)$.

(2) For the given parameter k and under the controller Eq. (5), the approximation error satisfies the following H^{∞} performance index

$$\frac{1}{2}\int_0^{T} e^{T} Q e\,dt \le \frac{1}{2} e^{T}(0) p e(0) + \sum_{i=1}^{m} \frac{1}{2\mu_i}\tilde{w}_{fi}^{T}(0)\tilde{w}_{fi}(0) + \sum_{i=1}^{m}\sum_{j=1}^{m} \frac{1}{2\rho_{ij}} \tilde{w}_{g_{ij}}^{T}(0)\tilde{w}_{g_{ij}}(0) + \frac{1}{2}\kappa^2 \int_0^{T} \omega^{T}\omega\,dt \qquad (25)$$

where $e = [e_{1p}^{T} \quad \cdots \quad e_{mp}^{T}]$, $Q = \operatorname{Diag}(Q_1 \quad \cdots \quad Q_m)$, $\omega = [\omega_1 \quad \cdots \quad \omega_n]^{T}$.

Proof[5, 6] (1) Suppose $V_{fi} = \frac{1}{2} w_{fi}^{T} w_{fi}$. When $\|w_{fi}\| \le M_{fi}$, from the first equation of Eq. (19) and the first equation of Eq. (23), its derivative along the system trajectory is given by

$$\dot{V}_f = -\sum_{i=1}^{m} \mu_i e_{ip}^{T} p_i B_i w_{fi}^{T}\Phi(x) + w_{fi}^{T}\epsilon(\|w_{fi}\| - M_{fi}) < 0 \qquad (26)$$

So $\|w_{fi}\| \le M_{fi}$ holds.

When $\|w_{fi}\| = M_{fi}$, from the second equation of Eq. (19), $\dot{V}_f = 0$, namely $\|w_{fi}\| \le M_{fi}$.

For any $t \ge 0$, there is $\|w_{fi}\| \le M_{fi}$. One can also prove $\|w_{fi}\| \le M_{fi}$ for $t \ge 0$.

(2) Determining

$$V_i = \frac{1}{2} e_{ip}^{T} p_i e_{ip} + \frac{1}{2\mu_i}\tilde{w}_{fi}^{T}\tilde{w}_{fi} + \sum_{j=1}^{m}\frac{1}{2\rho_{ij}}\tilde{w}_{g_{ij}}^{T}\tilde{w}_{g_{ij}} \qquad (27)$$

Choose $V = V_1 + V_2 + \cdots + V_m$ as a Lyapunov function to calculate the time derivative of V

$$\dot{V} = \dot{V}_1 + \dot{V}_2 + \cdots + \dot{V}_m \qquad (28)$$

where $\dot{V}_i$ is given by

$$\dot{V}_i = \frac{1}{2}\mathring{e}_{ip}^{T} p_i e_{ip} + \frac{1}{2} e_{ip}^{T} p_i \mathring{e}_{ip} +$$

$$\frac{1}{\mu_1}\dot{\tilde{w}}_{fi}^{\mathrm{T}}\tilde{w}_{fi} + \sum_{j=1}^{m}\frac{1}{\rho_{ij}}\dot{\tilde{w}}_{g_{ij}}^{\mathrm{T}}\tilde{w}_{g_{ij}} \quad (29)$$

For $\dot{\tilde{w}}_{fi} = \dot{w}_{fi}$, $\dot{\tilde{w}}_{g_{ij}} = \dot{w}_{g_{ij}}$, substituting Eqs. (17), (19) and Eq. (20) and Eq. (21) into Eq. (29) yields

$$\dot{V}_i = -\frac{1}{2}e_{ip}^{\mathrm{T}}Q_ie_{ip} - \frac{1}{2k^2}e_{ip}^{\mathrm{T}}p_iB_iB_i^{\mathrm{T}}p_ie_{ip} + \frac{1}{2}(\omega_i^{\mathrm{T}}B_i^{\mathrm{T}}p_ie_{ip} + e_{ip}^{\mathrm{T}}p_iB_i\omega_i) + (e_{ip}^{\mathrm{T}}p_iB_i\Phi_i^{\mathrm{T}}(x) + \frac{1}{\mu_i}\dot{w}_{fi}^{\mathrm{T}})\tilde{w}_{fi} + \sum_{j=1}^{m}((e_{ip}^{\mathrm{T}}p_iB_i\Phi_j^{\mathrm{T}}(x)u_j + \frac{1}{\rho_{ij}}\dot{w}_{g_{ij}}^{\mathrm{T}})\tilde{w}_{g_{ij}}) \quad (30)$$

It is not difficult to prove that the last two items in the above formula are negative.

It follows from

$$(\omega_i^{\mathrm{T}} - \frac{1}{\kappa^2}e_{ip}^{\mathrm{T}}p_iB_i)(\omega_i^{\mathrm{T}} - \frac{1}{\kappa^2}e_{ip}^{\mathrm{T}}p_iB_i)^{\mathrm{T}} \geq 0 \quad (31)$$

that

$$\dot{V}_i \leq -\frac{1}{2}e_{ip}^{\mathrm{T}}Q_ie_{ip} + \frac{1}{2}\kappa^2\omega_i^{\mathrm{T}}\omega \leq -\frac{1}{2}e_{ip}^{\mathrm{T}}Q_ie_{ip} + \frac{1}{2}\kappa^2|\bar{\omega}|^2 \quad (32)$$

where $\bar{\omega}$ is the upper bound of the compound disturbance.

Let $\lambda_{\min}(\cdot)$ denote the smallest eigenvalue of the matrix Q_i. When $\|e_{ip}\| > \frac{\kappa|\bar{\omega}|}{\lambda_{\min}(Q_i)}$, then

$$\dot{V}_i \leq -\frac{1}{2}\lambda_{\min}(Q_i)\|e_{ip}\|^2 + \frac{1}{2}\kappa^2|\bar{\omega}|^2 < 0 \quad (33)$$

which implies

$$\dot{V} = \dot{V}_1 + \dot{V}_2 + \cdots + \dot{V}_m < 0$$

So the close loop nonlinear system is stable under the designed controller.

Integrating Eq. (33) from $t=0$ to $t=T$ yields

$$V_i(T) - V_i(0) \leq -\frac{1}{2}\int_0^T e_{ip}^{\mathrm{T}}Q_ie_{ip}\mathrm{d}t + \frac{1}{2}\kappa^2\int_0^T\omega_i^{\mathrm{T}}\omega_i\mathrm{d}t \quad (34)$$

Eq. (26) implies $V_i(T) \geq 0$, and combined with Eq. (26) and Eq. (34), it yields

$$\frac{1}{2}\int_0^T e_{ip}^{\mathrm{T}}Q_ie_{ip}\mathrm{d}t \leq \frac{1}{2}e_{ip}^{\mathrm{T}}(0)p_ie_{ip}(0) + \frac{1}{2\mu_i}\tilde{w}_{fi}^{\mathrm{T}}(0)\tilde{w}_{fi}(0) + \sum_{j=1}^{m}\frac{1}{2\rho_{ij}}\tilde{w}_{g_{ij}}^{\mathrm{T}}(0)\tilde{w}_{g_{ij}}(0) + \frac{1}{2}\kappa^2\int_0^T\omega_i^{\mathrm{T}}\omega_i\mathrm{d}t \quad (35)$$

For the whole nonlinear system, there is

$$\frac{1}{2}\int_0^T e^{\mathrm{T}}Qe\mathrm{d}t \leq \frac{1}{2}e^{\mathrm{T}}(0)pe(0) + \sum_{i=1}^{m}\frac{1}{2\mu_i}\tilde{w}_{fi}^{\mathrm{T}}(0)\tilde{w}_{fi}(0) + \sum_{i=1}^{m}\sum_{j=1}^{m}\frac{1}{2\rho_{ij}}\tilde{w}_{g_{ij}}^{\mathrm{T}}(0)\tilde{w}_{g_{ij}}(0) + \frac{1}{2}\kappa^2\int_0^T\omega^{\mathrm{T}}\omega\mathrm{d}t \quad (36)$$

For the given parameter k, the designed H^{∞} controller reaches expected performances.

4 Simulation Study

The dynamics equations of the robot are described by the following differential equations

$$\begin{bmatrix}\ddot{q}_1\\ \ddot{q}_2\end{bmatrix} = \begin{bmatrix}H_{11} & H_{12}\\ H_{21} & H_{22}\end{bmatrix}^{-1}\begin{bmatrix}-h\dot{q}_2 & -h\dot{q}_1 - h\dot{q}_2\\ h\dot{q}_1 & 0\end{bmatrix}\cdot\begin{bmatrix}\dot{q}_1\\ \dot{q}_2\end{bmatrix} + \begin{bmatrix}H_{11} & H_{12}\\ H_{21} & H_{22}\end{bmatrix}^{-1}\begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix} + \begin{bmatrix}d_1\\ d_2\end{bmatrix} \quad (37)$$

where q_1 and q_2 are position vectors.

$$H_{11} = a_1 + 2a_3\cos(q_2) + 2a_4\sin(q_2)$$
$$H_{12} = H_{21} = a_2 + a_3\cos(q_2) + a_4\sin(q_2)$$
$$H_{22} = a_2,\ h = a_3\sin(q_2) - a_4\cos(q_2)$$

with

$$a_1 = I_1 + m_1l_{c_1}^2 + I_e + m_el_{c_e}^2 + m_el_1^2$$
$$a_2 = I_e + m_el_{c_e}^2,\ a_3 = m_el_1l_{c_e}\cos(\delta_e)$$
$$a_4 = m_el_1l_{c_e}\sin(\delta_e),\ d_1 = 0.5\sin(q_1)$$
$$d_2 = 0.5\sin(q_2)$$

and

$$l_1 = 1.0,\ l_{c_1} = 0.5,\ m_1 = 1.0,\ I_1 = 0.12$$
$$l_{c_e} = 0.6,\ \delta_e = \frac{\pi}{6},\ m_e = 2.0,\ I_e = 0.25$$

From the formula Eq. (37), one can see that the formula Eq. (37) is exactly the form of Eq. (3). The same neural networks are used to approximate f_1, f_2, g_{11}, $g_{12}(g_{21})$, g_{22}. The hidden layer of the RBF network has 15 entries. The initialization conditions are

$w_{g_{11}} = w_{g_{12}} = w_{g_{22}} = 0.5\boldsymbol{l}$; $w_{f1} = w_{f2} = \mathbf{0}$;
$q_1(0) = q_2(0) = 0.2$, $M_{f1} = M_{f2} = 10$;
$M_{f1} = M_{f2} = 10$; $M_{g_{11}} = M_{g_{12}} = M_{g_{22}} = 15$;
$\epsilon_1 = \epsilon_2 = 2\boldsymbol{l}$; $\sigma_{11} = \sigma_{12} = \sigma_{22} = 1.5\boldsymbol{l}$ ($\boldsymbol{l}$ is unit vector); the learning rates are $\mu_1 = \mu_2 = 0.1$, $\rho_{11} = \rho_{12} = \rho_{22} = 0.02$;

Fig. 1 and Fig. 2 present time plots of $k = 0.05, 0.2, 0.3$ and corresponding $\gamma = 0.004, 0.1,$

0.2. From Figs. 1 and 2, one can see that the stability track error will decrease when the k decreases.

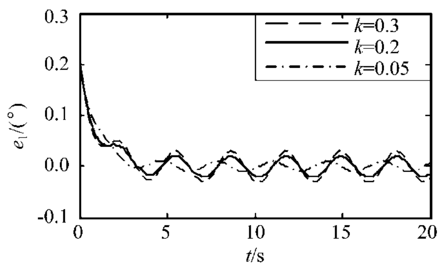

Fig. 1 Track error 1

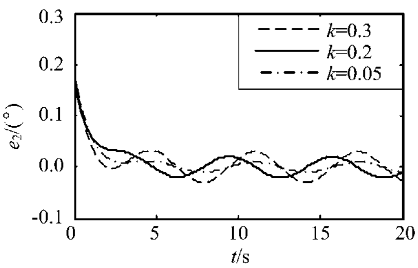

Fig. 2 Track error 2

5 Conclusions

The RBF neural networks combining adaptive control and robust control are used to control uncertain MIMO nonlinear systems. A new adaptive law has been proposed in this paper. The approximation errors of the neural networks are used in the adaptive law to improve performance of the systems. The simulation shows that tracking errors will be very small if the parameter k and matrix $\boldsymbol{Q}_i$ are correctly chosen.

References

[1] Hunt K J, Sbarbaro D. Neural networks for nonlinear internal model control[J]. IEEE Proceedings D, 1991, 138(5): 431-438.

[2] Park J, Sand Berg J W. Universal approximation using radial-basis-function networks [J]. Neural Computation, 1991, 3(2): 246-257.

[3] Chen S, Cowan C F N, Grant P M. Orthogonal least squares learning algorithm for radial basis function networks [J]. IEEE Trans Neural Networks, 1991, 2(2): 302-309.

[4] Sanner P, Slotine J J. Gaussian networks for direct adaptive control[J]. IEEE Transactions on Neural Networks 1992, 3(6), 837-864.

[5] Liu C C, Chen F H. Adaptive control of nonlinear continuous systems using neural networks-general relation degree and MIMO cases[J]. Int J of Control, 1993, 58(2): 317-335.

[6] Tong S C, Tang J T, Wang T. Fuzzy adaptive control of multivariable nonlinear systems[J]. Fuzzy Sets and Systems, 2000, 11(1), 163-167.

Biographies:

JIANG Chang-sheng Born in 1942, he is doyen Professor of Nanjing University of Aeronautics and Astronautics, doctoral advisor. His current research field is aerocraft control.

CHEN Mou Born in 1975, he received M.S. from Nanjing University of Aeronautics and Astronautics in 2001, and then he became a student of NUAA studying for Ph.D. in 2001. Now his research interests include uncertain nonlinear system control, integration fire/flight/propulsion control.